Unified Perspective: Theory, Practice, Interpretability, and Controllability of Diffusion Models with Applications in Medical Imaging, Inverse Problems, Signal Processing, Semantic Correspondence

Talha Ahmed
Rehan Ahmad
Nehal Ahmed Shaikh

Department of Electrical Engineering, LUMS

This survey attempts at a comprehensive exploration of recent advancements in diffusion models especially focused on its theory, interpretability and control mechanisms. We intend to show its applications across medical imaging, inverse problems, signal processing, and semantic correspondence with insights drawn from current methodologies and innovations.

Contents

1 Introduction

1.1 Background and Relevance

Diffusion models have emerged as a powerful class of generative models, effectively addressing complex tasks across medical imaging, signal processing, and various inverse problems. This survey aims to bridge theoretical insights with practical applications in these domains, emphasizing the adaptability of diffusion models in high-stakes fields requiring both interpretability and controllability.

1.2 Structure of the Survey

This survey is organized as follows:

- Section 2 presents foundational concepts in diffusion models, including basic principles, key algorithms and variants, classifier guidance, plug-and-play methods, and score-based models.
- Section 3 provides an overview of inverse problems and image restoration, covering super-resolution, inpainting, turbulence, blurring, and phase retrieval.
- Section 4 focuses on key diffusion model algorithms for image restoration, featuring notable approaches such as Plug-and-Play Image Restoration, StableSR, DiffBR, BIRD, DPS, Blind-DPS, ReSample, and RePaint.
- Section 5 discusses challenges in medical imaging with diffusion models, including high dimensionality and stable inference.
- Section 6 examines conditional and interpretable image segmentation, highlighting controllability in text-image diffusion models and techniques for multi-class segmentations in medical imaging.
- Section 7 explores knowledge distillation in diffusion models, detailing methods like Plug-and-Play Diffusion Distillation, Adversarial Diffusion Distillation, and Conditional Diffusion Distillation (CoDi).
- Section 8 investigates adapting diffusion models for various domains, focusing on discrete, invariant, and low-rank structures.
- Section 9 concludes with controllable and semantically consistent generation, discussing techniques for ensuring semantic consistency and their applications in semantic correspondence.

2 Foundational Concepts in Diffusion Models

2.1 Diffusion Models Basics

An introduction to the principles of diffusion modeling, discussing concepts like the forward and reverse processes, training objectives, and their theoretical foundation.

2.2 Key Algorithms and Variants

Overview of core algorithms such as DDPM and DDIM, including their mathematical formulations.

2.3 Classifier Guidance and Classifier Free Guidance

2.4 Plug-and-Play Methods

2.5 Score-Based Models and Flow Matching

Introduction to score-based modeling and flow matching techniques, highlighting their applications in generating data distributions with complex structures. Also providing an equivalent score-based formulation of DDPM/DDIM

2.6 Architectural Details

Discuss the foundational architecture introduced in the paper Denoising Diffusion Probabilistic Models (DDPM). Then talk about the U-NeT architecture. There was a good visualization given in the survey, "Diffusion Models and Representation Learning - a Survey". Incorporate insights from that paper. Then transition by talking about DiT.

2.6.1 Diffusion Transformers

Peebles and Xie [26] introduced the Diffusion Transformer (DiT) and demonstrated its potential of transformers in achieving state-of-the-art image generation with diffusion processes. Similar to the U-NeTs, DiT operate in the VAE's latent space.

To begin with, an image I is encoded to its latent representation $\vec{z}$ using VAE (for I of shape $256 \times 256 \times 3$, the latent vector $\vec{z}$ is $32 \times 32 \times 4$) The first layer of DiT is *patchify*, which converts the spatial input into a sequence of $T = (I/p)^2$ tokens with an embedding dimension d. (for patches of size $4 \times 4 \times 4$, $\vec{z} = 32 \times 32 \times 4$, $I = 256 \times 256 \times 3$, we obtain 64 tokens each of size 64)

This is embedded in a hidden dimension d using a weight matrix E, resulting in patches $[\vec{p}_1, \ldots, \vec{p}_T]$ where $\vec{p}_i \in \mathbb{R}^{1 \times d}$. The authors add fixed sinusoidal positional embeddings which serve to encode positional information without being modified during

training. Let $E_{\text{pos}} \in \mathbb{R}^{1\times d}$, then the initial latent representation can be written as:

$$\vec{z}_0 = [\vec{p}^1 + E^1_{\text{pos}}; \vec{p}^2 + E^2_{\text{pos}}; \ldots; \vec{p}^T + E^T_{\text{pos}}] \in \mathbb{R}^{T\times d} \tag{1}$$

Subsequently, *timestep Embedding* are incorporated for each timestep t. A sinusoidal embedding encodes temporal information in a way that allows the model to easily distinguish between different timesteps. It maps a scaled timestep t into a high-dimensional vector using sinusoidal functions. From the official code associated with the paper, the relevant line of computation that calculate the embeddings is:

$$\text{freqs}[k] = \exp\left(-\frac{\log(\text{max\_period})\cdot k}{d/2}\right) \tag{2}$$

where k is an index that ranges from 0 to $\frac{d}{2}-1$. These frequencies are then multiplied by the timestep t to obtain the argument values

$$\text{args} = t \cdot \text{freqs} \tag{3}$$

.

The final embedding vector is formed by concatenating the cosine and sine values of the arguments, resulting in an embedding of dimension d:

$$t = [\cos(\text{args}), \sin(\text{args})] \tag{4}$$

For example, if timestep $t = 50$, embedding dimension $d = 8$, max_period $= 10000$, then $\text{freqs}[k] = \exp([0, -2.30, -4.6, -6.9])$. This gives freqs $= [1, 0.1, 0.01, 0.001]$ We then calculate args $= 50 \cdot [1, 0.1, 0.01, 0.001] = [50, 5, 0.5, 0.5]$ Finally, we can calculate the embedding at time step $t = 50$ using equation (4).

In the next step, the authors introduce *label embedding* . Given a batch of labels $\vec{y}$ where $\vec{y}$ is a one-hot encoded vector, $\vec{c}_i = E[\vec{y}_i]$. Here, $E \in \mathbb{R}^{T\times d}$ where T is the sequence length. Note that for a single latent vector, there would only be a single label embedder associated with it. The final conditioning information then becomes

$$(\text{seq})_{\text{con}} = [\vec{c}_i, t] \tag{5}$$

The authors explore four options to incorporate the conditioning information in (5) with the patches obtained from (1):

- They append the vector embeddings at time t, $\vec{z}_t$, and $(\text{seq})_{\text{con}} = [\vec{c}_i, t]$ as two additional tokens in the input sequence. This treats the tokens as no differently from the image tokens. This is similar to `cls` tokens in ViTs and it allows us to use standard ViT blocks without modification. After the final block, we remove the conditioning tokens from the sequence. This approach introduces negligible new Gflops to the model.

- They concatenate the embeddings of t and $\vec{c}$ into a length-two sequence, separate

from the image token sequence $\vec{z}_l^t$. The transformer block is modified to include an additional multi-head cross-attention layer. Thus,

$$(\text{seq})_{\text{con}} = [\vec{c}_i, t] \in \mathbb{R}^{2\times d} \tag{6}$$

where d is the embedding dimension. The image token $\vec{z}_l^t$ attends to the conditioning token $\vec{t}$ and $\vec{y}_i$. The cross-attention mechanism is expressed as:

$$\text{attn}(\vec{z}_l^t, (\text{seq})_{\text{con}}) = \text{softmax}\left(\frac{Q(\vec{z}_l^t)K((\text{seq})_{\text{con}})^T}{\sqrt{d}}\right)V((\text{seq})_{\text{con}}) \tag{7}$$

- They explore replacing the standard layer normalization in transformer blocks with adaptive layer normalization (AdaLN). Rather than directly learning dimension-wise scale and shift parameters γ and β, these parameters are regressed from the sum of the embedding vectors of t and $\vec{c}_i$. In the standard layer normalization, we have $\hat{x} = \frac{\vec{x}-\mu}{\sigma} \cdot \gamma + \beta$ where μ is the mean and σ is the standard deviation, and γ and β are learned scale and shift parameters. Since this approach is the most compute-efficient, the authors choose it as their conditioning method.

- AdaLN-Zero: In addition to γ and β, AdaLN-Zero introduces a scaling factor α that is also regressed from the same conditioning information. Given an input $\vec{x}$, it is first normalized using AdaLN: $\hat{x} = \frac{\vec{x}-\mu}{\sigma} \cdot \gamma + \beta$. Before applying the residual connection, the output of the block is scaled by the zero-initialized α parameter: $\vec{y} = \alpha f(\hat{x})$ where $f(\hat{x})$ represents the operations performed within the residual block (e.g., convolutions, additional transformations). Finally, the input $\vec{x}$ is added back to the scaled output $\vec{y}$ such that $\vec{z} = \vec{x} + \vec{y} = \vec{x} + \alpha f(\hat{x})$. Since α is initially zero, the output $\vec{z}$ is just the input $\vec{x}$, making the block behave as an identity function initially.

The subsequent *Transformer Block with AdaLN* retains much of its similarity with ViT except for the addition of conditioning information using Ada-LN Zero:

$$\vec{z}_l = \text{FFN}\left\{\text{Attn}\left\{\text{AdaLN}(\vec{z}_{l-1})\right\} + \alpha_z \text{AdaLN}(\vec{z}_{l-1})\right\}, \quad \vec{z}_l \in \mathbb{R}^{T\times d} \tag{8}$$

After passing through l layers of the Transformer Block using AdaLN the final tensor is *unpatchified* into the shape (h, w, p, p, c). This is reshaped to $(c, h \cdot p, w \cdot p)$. Thus, the output vector at timestep t is of the same dimensionality as the input vector at timestep t. This vector $\vec{z}^t$ is then used to estimate the mean μ and Σ associated at time step t and serves as input for $t + 1$ step.

Esser et al [8] introduces DiTs in latent space for text-to-image generation which further cements the architecture's increasing popularity in mainstream computer Vision. In their work, a text input is first encoded using CLIP G/14, CLIP L/14, or T5 XXL into a sequence of embeddings $c_{\text{txt}} \in \mathbb{R}^{(n\times d_{\text{txt}})}$ where n is the number of tokens associated with the text, and d_{txt} is the dimensionality of the text embedding.

The text embeddings are then projected into the same dimensionality d as the image patch embedding. Positional encodings are added to the text embeddings. At this point, we have n text embeddings:

$$\vec{c}_t = [\vec{t}_1, \ldots, \vec{t}_n] \in \mathbb{R}^{(n \times d)} \tag{9}$$

We concatenate these with T spatial patches of dimension d:

$$\vec{z}_t = [\vec{p}_1, \ldots, \vec{p}_T] \in \mathbb{R}^{(T \times d)} \tag{10}$$

Thus,

$$\text{Conc\_seq} = [\vec{p}_1, \ldots, \vec{p}_T; \vec{t}_1, \ldots, \vec{t}_n] \in \mathbb{R}^{((n+T) \times d)} \tag{11}$$

Throughout the Stable Diffusion 3 pipeline, different sets of weights are utilized for the two modalities. However the tokens are joined for the attention operation. This allows both representations to work in their own space while taking the other into account. The conditioning information $\vec{y} + \vec{t} \in \mathbb{R}^{(1 \times d)}$ is once again regressed by an MLP to obtain shift and scale parameters α_c, β_c, α_x, and β_x. Both of these modalities go through distinct Ada-LN DiT blocks:

$$\begin{aligned} \vec{z}_l &= \text{FFN}\left\{\text{Attn}\left\{\text{AdaLN}(\vec{z}_{l-1})\right\} + \alpha_z \text{AdaLN}(\vec{z}_{l-1})\right\} \\ \vec{c}_l &= \text{FFN}\left\{\text{Attn}\left\{\text{AdaLN}(\vec{c}_{l-1})\right\} + \alpha_c \text{AdaLN}(\vec{c}_{l-1})\right\} \end{aligned} \tag{12}$$

The weights that generate query Q, key K, and value V for these two modalities are shared:

$$Q = [\vec{z}_l; \vec{c}_l]W_Q, \quad K = [\vec{z}_l; \vec{c}_l]W_K, \quad V = [\vec{z}_l; \vec{c}_l]W_V \tag{13}$$

where W_Q, W_K, and $W_V \in \mathbb{R}^{(d \times d)}$ are learned projections. The attention is subsequently computed as:

$$\text{Attention}(Q_i, K_i, V_i) = \text{softmax}\left(\frac{Q_i K_i^T}{\sqrt{d_k}}\right) V_i \tag{14}$$

Afterward, the concatenation is undone to retrieve the spatial patches $\vec{z}_l$ and the textual patches $\vec{c}_l$. Each of the modalities is passed through separate FFN networks. Note that the only weights that are shared are those that generate the Query Q, Key K, and Values V. Otherwise, the weights of each modality are entirely distinct. The pipeline of SD3, taken from the original paper, is shown below.

3 Overview of Inverse Problems and Image Restoration

3.1 Super Resolution

3.2 Inpainting

3.3 Turbulence

3.4 Blurring

3.5 Phase Retrieval

4 Key Diffusion Model Algorithms for Inverse Problems and Image Restoration

4.1 Plug-and-Play Image Restoration

4.2 StableSR, DiffBR, BIRD, DPS, and Blind-DPS, ReSample, RePaint

Detailed analysis of specific diffusion algorithms and their application to inverse problems, with a focus on BIRD, DPS, and Blind-DPS.

4.3 Key Takeaways

Discussion of the ReSample method and other recent contributions to diffusion-based image restoration and enhancement.

5 Challenges in Medical Imaging with Diffusion Models

5.1 Addressing High Dimensionality and Stable Inference

Methods for handling high-dimensional data in diffusion models and strategies for effective image representation in medical imaging.

5.2 Cross-Modality Learning and Latent Space Optimization

Survey of cross-modality learning strategies and latent space optimization methods in diffusion models for improved adaptability and accuracy.

6 Conditional and Interpretable Image Segmentation

6.1 Overview of Controllability in Text-Image Diffusion Models

6.2 Techniques for Multi-Class Segmentations

Discussion on segmentation methods for diverse medical imaging tasks, such as Cardiac MRI, and the application of attention mechanisms.

6.3 Conditional Image Segmentation in Medical Imaging

.

Overview of techniques for conditional segmentation, including multi-class segmentation for medical imaging applications.

7 Knowledge Distillation in Diffusion Models

7.1 Plug-and-Play Diffusion Distillation

7.2 Adverserial Diffusion Distillation

7.3 CoDi: Conditional Diffusion Distillation

8 Adapting Diffusion Models for Various Domains

8.1 Discrete, Invariant, Manifold Structural Data

.

8.2 Low-Rank and Sparse Structure

Techniques for incorporating low-rank and sparse structures in diffusion models for signal processing, highlighting control strategies and model modifications.

9 Controllable and Semantically Consistent Generation

9.1 Techniques for Semantic Consistency

Strategies to achieve semantic consistency in generated images, including attention maps and embeddings for meaningful generation.

9.2 Semantic Correspondence Applications

Applications of semantically consistent generation, with a focus on unsupervised learning and semantic correspondence tasks.

10 Conclusion

10.1 Summary of Key Findings

A summary of the key insights from the survey and their implications for diffusion models in medical imaging and signal processing.

10.2 Limitations and Open Challenges

Discussion of current limitations and potential challenges for future research in these domains.

A Appendices

A.1 Code Resources

Links to relevant GitHub repositories, tools, or resources.

A.2 Suplementary Material

Supplemental proofs and derivations supporting the discussed methodologies.

A.2.1 Tweedie's Formula

A.2.2 Total Variation Regularization

A.3 Additional Figures

Figures and diagrams for extended illustration of key concepts.

Acknowledgments

This work has been supported and guided by Dr. Hassan Mohy-ud-Din and Dr. Muhammad Tahir. Their mentorship and expertise have been instrumental in developing this research.

References

[1] Pu Cao, Feng Zhou, Qing Song, and Lu Yang. Controllable generation with text-to-image diffusion models: A survey, 2024.

[2] Hamadi Chihaoui, Abdelhak Lemkhenter, and Paolo Favaro. Blind image restoration via fast diffusion inversion, 2024.

[3] Hyungjin Chung, Jeongsol Kim, Sehui Kim, and Jong Chul Ye. Parallel diffusion models of operator and image for blind inverse problems, 2022.

[4] Hyungjin Chung, Jeongsol Kim, Michael T. Mccann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems, 2024.

[5] Hyungjin Chung, Suhyeon Lee, and Jong Chul Ye. Decomposed diffusion sampler for accelerating large-scale inverse problems, 2024.

[6] Mischa Dombrowski, Hadrien Reynaud, Matthew Baugh, and Bernhard Kainz. Foreground-background separation through concept distillation from generative image foundation models, 2023.

[7] Weisheng Dong, Peiyao Wang, Wotao Yin, Guangming Shi, Fangfang Wu, and Xiaotong Lu. Denoising prior driven deep neural network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(10):2305–2318, October 2019.

[8] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*, 2024.

[9] Alexandros Graikos, Nikolay Malkin, Nebojsa Jojic, and Dimitris Samaras. Diffusion models as plug-and-play priors, 2023.

[10] Eric Hedlin, Gopal Sharma, Shweta Mahajan, Hossam Isack, Abhishek Kar, Andrea Tagliasacchi, and Kwang Moo Yi. Unsupervised semantic correspondence using stable diffusion, 2023.

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020.

[12] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022.

[13] Yi-Ting Hsiao, Siavash Khodadadeh, Kevin Duarte, Wei-An Lin, Hui Qu, Mingi Kwon, and Ratheesh Kalarot. Plug-and-play diffusion distillation, 2024.

[14] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models, 2022.

[15] Ivan Kobyzev, Simon J.D. Prince, and Marcus A. Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):3964–3979, November 2021.

[16] Benedikt Kolbeinsson and Krystian Mikolajczyk. Multi-class segmentation from aerial views using recursive noise diffusion, 2024.

[17] Xin Li, Yulin Ren, Xin Jin, Cuiling Lan, Xingrui Wang, Wenjun Zeng, Xinchao Wang, and Zhibo Chen. Diffusion models for image restoration and enhancement – a comprehensive survey, 2023.

[18] Tianyu Lin, Zhiguang Chen, Zhonghao Yan, Weijiang Yu, and Fudan Zheng. Stable diffusion segmentation for biomedical images with single-step reverse process, 2024.

[19] Xinqi Lin, Jingwen He, Ziyan Chen, Zhaoyang Lyu, Bo Dai, Fanghua Yu, Wanli Ouyang, Yu Qiao, and Chao Dong. Diffbir: Towards blind image restoration with generative diffusion prior, 2024.

[20] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling, 2023.

[21] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models, 2022.

[22] Calvin Luo. Understanding diffusion models: A unified perspective, 2022.

[23] Michael T. McCann, Hyungjin Chung, Jong Chul Ye, and Marc L. Klasky. Score-based diffusion models for bayesian image reconstruction, 2023.

[24] Kangfu Mei, Mauricio Delbracio, Hossein Talebi, Zhengzhong Tu, Vishal M. Patel, and Peyman Milanfar. Codi: Conditional diffusion distillation for higher-fidelity and faster image generation, 2024.

[25] Andrew S. Na, William Gao, and Justin W. L. Wan. Efficient denoising using score embedding in score-based diffusion models, 2024.

[26] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

[27] Dominic Rampas, Pablo Pernias, and Marc Aubreville. A novel sampling scheme for text- and image-conditional image synthesis in quantized latent spaces, 2023.

[28] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation, 2023.

[29] Bowen Song, Soo Min Kwon, Zecheng Zhang, Xinyu Hu, Qing Qu, and Liyue Shen. Solving inverse problems with latent diffusion models via hard data consistency, 2024.

[30] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022.

[31] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2021.

[32] Lingchen Sun, Rongyuan Wu, Jie Liang, Zhengqiang Zhang, Hongwei Yong, and Lei Zhang. Improving the stability and efficiency of diffusion models for content consistent super-resolution, 2024.

[33] Jianyi Wang, Zongsheng Yue, Shangchen Zhou, Kelvin C. K. Chan, and Chen Change Loy. Exploiting diffusion prior for real-world image super-resolution, 2024.

[34] Sven-Ake Wegner. Lecture notes on high-dimensional data, 2024.

[35] Julia Wolleb, Robin Sandkühler, Florentin Bieder, Philippe Valmaggia, and Philippe C. Cattin. Diffusion models for implicit image segmentation ensembles, 2021.

[36] Di Wu, Shicai Fan, Xue Zhou, Li Yu, Yuzhong Deng, Jianxiao Zou, and Baihong Lin. Unsupervised anomaly detection via masked diffusion posterior sampling, 2024.

[37] Junde Wu, Wei Ji, Huazhu Fu, Min Xu, Yueming Jin, and Yanwu Xu. Medsegdiff-v2: Diffusion based medical image segmentation with transformer, 2023.

[38] Yanwu Xu, Li Sun, Wei Peng, Shuyue Jia, Katelyn Morrison, Adam Perer, Afrooz Zandifar, Shyam Visweswaran, Motahhare Eslami, and Kayhan Batmanghelich. Medsyn: Text-guided anatomy-aware synthesis of high-fidelity 3-d ct images. *IEEE Transactions on Medical Imaging*, 43(10):3648–3660, 2024.

[39] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications, 2024.

[40] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte. Plug-and-play image restoration with deep denoiser prior, 2021.

[41] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023.